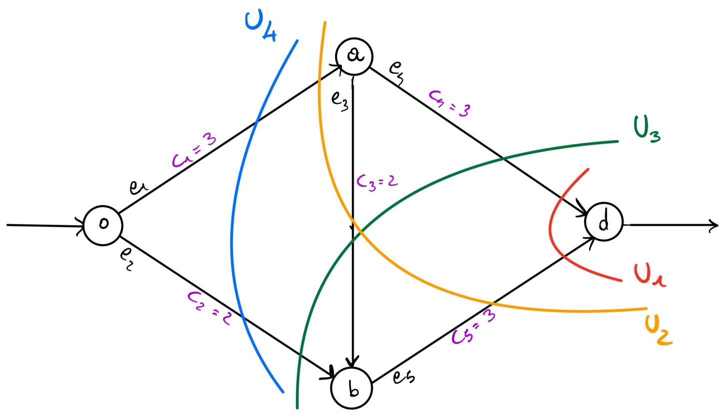
# Homework 1

Giulio Nenna (s245717@studenti.polito.it)

November 18, 2020

**Esercizio 1.**




Tutti i possibili path $o - d$ sono i seguenti:

**1.** $p_{(1)} = o - a - d$

**2.** $p_{(2)} = o - b - d$

**3.** $p_{(3)} = o - a - b - d$

I possibili cut e annesse capacità sono:

$$
\begin{aligned}
U_1 &= \{d\} & C(U_1) &= 5 \\
U_2 &= \{a, d\} & C(U_2) &= 5 \\
U_3 &= \{b, d\} & C(U_3) &= 7 \\
U_4 &= \{a, b, d\} & C(U_4) &= 5
\end{aligned}
$$

**(a)** Tutti i possibili colli di bottiglia del grafo hanno capacità pari a 5, quindi per ostruire qualsiasi tipo di flow unitario da $o$ a $d$ bisogna eliminare una capacità $C^*$ tale che

$$C^* < 4$$

**(b)** Dal momento che vale il teorema del *Max Flow - Min Cut*, le due unità di capacità devono essere distribuite in modo tale da aumentare lacapacità del minimo taglio. Poiché ci sono ben tre tagli su 4 che rappresentano un minimo taglio $(U_1, U_2, U_4)$ è necessario aumentare la capacità su archi in comune a questi tre tagli.
Una scelta possibile potrebbe essere quella di aggiungere un'unità di capacità ciascuno agli archi $e_1$ ed $e_4$ in modo tale che i tagli risultino:

$$\begin{aligned} U_1 &= \{d\} & C(U_1) &= 6 \\ U_2 &= \{a, d\} & C(U_2) &= 6 \\ U_3 &= \{b, d\} & C(U_3) &= 9 \\ U_4 &= \{a, b, d\} & C(U_4) &= 6 \end{aligned}$$

Aumentando in questo modo il massimo troughput da 5 a 6

**(c)** Assegnati i seguenti delay a ciascun arco:

$$d_1(x) = d_5(x) = x + 1, \quad d_3(x) = 1, \quad d_2(x) = d_4(x) = 5x + 1$$

e definendo con $z_i$ il flusso sull'i-esimo path, si calcola il delay complessivo su ciascun path:

$$\begin{aligned} d_{p_1} &= 6z_1 + z_3 + 2 \\ d_{p_2} &= 6z_2 + z_3 + 2 \\ d_{p_3} &= 2z_3 + z_1 + z_2 + 3 \end{aligned}$$

Si calcola ora l'equilibrio di Wardrop considerando che $z_1 + z_2 + z_3 = 1$ (quindi che $z_1 + z_2 = 1 - z_3$)e che il delay su ciascun path con flusso non nullo deve essere minore di qualsiasi altro delay.

$$z_1 < 0 \Rightarrow \quad 6z_1 + z_3 + 2 \leq 2z_3 + z_1 + z_2 + 3, \quad \Rightarrow z_1 \leq \frac{1}{3}$$

$$
\begin{aligned}
z_3 > 0 \Rightarrow \qquad & 2z_3 + z_1 + z_2 + 3 \leq 6z_1 + z_3 + 2 && \Rightarrow z_1 \geq \frac{1}{3} \\
& 2z_3 + z_1 + z_2 + 3 \leq 6z_2 + z_3 + 2 && \Rightarrow z_2 \geq \frac{1}{3}
\end{aligned}
$$

$$
z_2 > 0 \Rightarrow \qquad 6z_2 + z_3 + 2 \leq 2z_3 + z_1 + z_2 + 3 \qquad \Rightarrow z_2 \leq \frac{1}{3}
$$

Queste condizioni implicano che un possibile vettore di flusso che soddisfi le condizioni di Wardrop sia il seguente:

$$
z_{\text{Wardrop}} = z_W = \left(\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right)
$$

Il ritardo totale che deriva da questo vettore di flusso è dato da:

$$
\frac{2}{3}\left(\frac{2}{3}+1\right) + \frac{5}{9} + \frac{1}{3} + \frac{1}{3} + \frac{5}{9} + \frac{1}{3} + \frac{2}{3}\left(\frac{2}{3}+1\right) = \frac{13}{3}
$$

**(d)** Calcolare l'ottimo di sistema equivale a minimizzare il ritardo totale in funzione del flusso su ciascun path $o$ - $d$:

$$
\min_z (z_1+z_3)(z_1+z_3+1)+(5z_2^2+z_2)+(z_3)+(5z_1^2+z_1)+(z_2+z_3)(z_2+z_3+1)
$$

Utilizzando la condizione di flusso unitario ($z_1+z_2+z_3=1$) si ottiene:

$$
\begin{aligned}
&\min_z (1-z_2)(2-z_2) + (5z_2^2+z_2) + \\
&+(1-z_1-z_2) + (5z_1^2+z_1) + (1-z_1)(2-z_2) = \\
&\min_z 3(2z_1^2 - z_1) + 3(2z_2^2 - z_2) + 5
\end{aligned}
$$

Da cui si deduce:

$$
z_1 = \frac{1}{4} \qquad\qquad z_2 = \frac{1}{4}
$$

Quindi:

$$
z_{\text{Sistema}} = z_S = \left(\frac{1}{4}, \frac{1}{4}, \frac{1}{2}\right)
$$

Questa configurazione dei flussi produce un ritardo totale dato da:

$$\frac{3}{4}\left(\frac{3}{4}+1\right)+\frac{1}{4}\left(\frac{5}{4}+1\right)+\frac{1}{2}+\frac{1}{4}\left(\frac{5}{4}+1\right)+\frac{3}{4}\left(\frac{3}{4}+1\right)=\frac{17}{4}$$

**(e)** Il costo totale dell'anarchia è dato dal rapporto tra il il ritardo totale dell'ottimo di sistema e il ritardo totale dell'equilibrio di Wardrop

$$\frac{\frac{13}{3}}{\frac{17}{4}}=\frac{52}{51}$$

**(f)** Siano $f^*_{e_i}$ e $f^{(w)}_{e_i}$ i flussi su ciascun arco $e_i$ rispettivamente nella configurazione dell'ottimo di sistema e in quella di equilibrio di Wardrop. Sia il ritardo totale su ciascun arco $e_i$ definito come $C_{e_i}(f_e) = d_{e_i} f_{e_i}$. La configurazione ottimale dei pedaggi per ottenere un costo dell'anarchia unitario è la seguente:

$$\omega_{e_i} = C'_{e_i}(f^*_{e_i}) - d_{e_i}(f^*_{e_i}) = f^*_{e_i} d'_{e_i}(f^*_{e_i})$$

che nel caso specifico si traduce in:

$$\begin{aligned}
\omega_{e_1} &= \frac{3}{4}1 = \frac{3}{4}\\
\omega_{e_2} &= \frac{1}{4}5 = \frac{5}{4}\\
\omega_{e_3} &= 0\\
\omega_{e_4} &= \frac{5}{4}\\
\omega_{e_4} &= \frac{3}{4}
\end{aligned}$$

**Esercizio 2.**

**(a)** La matrice dei pesi $W$ è la seguente:

$$W=\begin{bmatrix} a & 1 & 0 & 0\\ 0 & 0 & 1 & 1\\ 0 & 0 & 0 & 1\\ 1 & 1 & 0 & 0\end{bmatrix}$$

la matrice dei pesi normalizzata è data da:

$$\omega = \begin{pmatrix} a+1 \\ 2 \\ 1 \\ 2 \end{pmatrix} \qquad \Rightarrow \qquad P = \begin{bmatrix} \frac{a}{a+1} & \frac{1}{a+1} & 0 & 0 \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 0 & 1 \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 \end{bmatrix}$$

Di conseguenza il laplaciano è dato da:

$$L = D - W = \begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 2 & -1 & -1 \\ 0 & 0 & 1 & -1 \\ -1 & -1 & 0 & 2 \end{bmatrix}$$

**(b)** La dinamica di opinione di French-De Groot è data da:

$$x(t+1) = Px(t)$$

dove $x(t) = (x_1(t), x_2(t), x_3(t), x_4(t))$ e $x_i(t)$ rappresenta il valore dell'opinione del nodo $i$ al tempo $t$.

**(c)** Affinchè l'opinione su $\mathcal{G}$ converga ad un consenso il grafo deve essere fortemente connesso ed aperiodico. Date le ridotte dimensioni del grafo è possibile verificare queste due proprietà manualmente, notando che è possibile raggiungere qualsiasi nodo a partire da qualsiasi posizione sul grafo e che ad esempio il nodo 4 ha un ciclo di lunghezza 2 e un altro di lunghezza 3. Connessione ed aperiodicità sono indipendenti dal valore di $a$ pertanto vale:

$$\lim_{t\to\infty} x(t) = \pi' x(0) \qquad\qquad \forall a \geq 0$$

Dal momento che $\mathcal{G}$ è bilanciato la misura di probabilità invariante $\pi$ è proporzionale al vettore $\omega$. In particolare:

$$\pi = \frac{\omega}{\mathbb{1}'\omega} = \begin{pmatrix} \frac{a+1}{a+6} & \frac{2}{a+6} & \frac{1}{a+6} & \frac{2}{a+6} \end{pmatrix}'$$

Pertanto il valore del consenso converge ad un valore finito fintanto che $a \neq -6$, condizione automaticamente verificata dal momento che $a \geq 0$

**(d)** Nel caso particolare in cui $a = 0$ si ha:

$$\pi = \begin{pmatrix} \frac{1}{6} & \frac{1}{3} & \frac{1}{6} & \frac{1}{3} \end{pmatrix}'$$

Per cui, date le condizioni iniziali $x(0)$, il consenso raggiunge il seguente valore:

$$\lim_{t\to\infty} x_i(t) = \pi' x(0) = \frac{1}{3} \qquad \forall i = 1 \dots 4$$

**(e)** Poiché il consenso viene raggiunto $\forall a \geq 0$, date le condizioni iniziali $x(0)$, vale la seguente:

$$\lim_{t\to\infty} x_1(t) = \pi' x(0) = \frac{1}{a+6}(-a+1+2-1+2) = \frac{2-a}{a+6}$$

segue quindi che:

$$\lim_{t\to\infty} x_1(t) \leq 0 \qquad \Rightarrow \qquad \frac{2-a}{a+6} \leq 0 \qquad \Rightarrow \qquad a \geq 2$$

**(f)** Date $x_i(0)$ variabili aleatorie indipendenti e identicamente distribuite tali che $Var(x_i(0)) = 1 \ \forall i = 1 \dots 4$, vale la seguente:

$$\lim_{t\to\infty} x_1(t) = \frac{a+1}{a+6}x_1(0) + \frac{2}{a+6}x_2(0) + \frac{1}{a+6}x_3(0) + \frac{2}{a+6}x_4(0)$$

Poiché le variabili sono i.i.d. e $Var(x_i(0)) = 1 \ \forall i = 1 \dots 4$ vale:

$$\begin{aligned} Var\left(\lim_{t\to\infty} x_1(t)\right) &= \left(\frac{a+1}{a+6}\right)^2 Var(x_1(0)) + \left(\frac{2}{a+6}\right)^2 Var(x_2(0)) + \\ &+ \left(\frac{1}{a+6}\right)^2 Var(x_3(0)) + \left(\frac{2}{a+6}\right)^2 Var(x_4(0)) = \\ &= \frac{a^2+2a+10}{(a+6)^2} = v(a) \end{aligned}$$

Essendo $v(a)$ una funzione convessa, si verifica facilmente che raggiunge un minimo per:

$$a = \frac{4}{5}$$

Che rappresenta il valore di $a$ per cui viene minimizzata la varianza del consenso.

**Esercizio 3.**

**(a)** Dal momento che il grafo in analisi è un grafo fortemente conneso e aciclico la dinamica di French-De Groot convergerà ad un consenso come visto per il secondo esercizio. Poichè il grafo è undirected la misura di probabilità invariante è proporzionale al vettore $\omega$, quest'ultimo è facilmente calcolabile in quanto corrisponde in questo caso al vettore dei gradi uscenti di ogni nodo. In particolare:

$$\omega = (1,3,3,4,3,4,2,2,3,1,3,6,1,2,1)$$
$$\pi = \frac{1}{38}\omega$$

quindi dal momento che il vettore degli stati iniziali è dato da:

$$x(0) = (0,0,0,0,0,-1,0,0,0,0,0,1,0,0,0)$$

il valore del consenso sarà:

$$\lim_{t\to\infty} x_i(t) = \left(\pi' x(0)\right)_i = \frac{6-4}{38} = \frac{1}{19} \qquad \forall i = 1\dots 15$$

**(b)** Il seguente codice MATLAB produce una simulazione della dinamica di French-De Groot assegnando i due nodi stubborn $\{\text{Strozzi}, \text{Medici}\}$ e fornendo una rappresentazione grafica della dinamica tramite la colorazione progressiva dei nodi in funzione della loro opinione

```
    close all
clear all
clc
%Costruzione della matrice di adiacenza
W = zeros (15);
W(1,12)=1; W(12,1)=1; W(3,2)=1; W(2,3)=1; W(13,2)=1;
W(2,13)=1; W(14,2)=1; W(2,14)=1; W(12,3)=1; W(3,12)=1;
W(14,3)=1; W(3,14)=1; W(12,4)=1; W(4,12)=1; W(4,15)=1;
W(15,4)=1; W(5,14)=1; W(14,5)=1; W(5,15)=1; W(15,5)=1;
W(6,7)=1; W(7,6)=1; W(6,12)=1; W(12,6)=1; W(15,6)=1;
W(6,15)=1; W(13,8)=1; W(8,13)=1; W(15,8)=1; W(8,15)=1;
W(9,15)=1; W(15,9)=1; W(11,10)=1; W(10,11)=1; W(10,15)=1;
W(15,10)=1; W(13,14)=1; W(14,13)=1;


Names={'Lamberteschi', 'Peruzzi', 'Bischeri','Tornabuoni',
```

```
 'Ridolfi', 'Albizzi', 'Ginori', 'Barbadori',
  'Acciaiuoli', 'Salviati', 'Pazzi', 'Guadagni',
   'Castellani', 'Strozzi', 'Medici'}';
index=[1:15]';
x=zeros(15,1); %Vettore di opinioni

%Inserisco i due nodi stubborn
x(15)=1;
x(14)=-1;

%Costruzione del Grafo in un oggetto graph
NodeTable=table(index, Names, x); %Informazioni sui nodi
G=graph(W,NodeTable);

%------------------Plot del grafo iniziale--------------------------

%Colorazione dei nodi
color=zeros(15,3);
for i=1:15
    if x(i)>0
        color(i,1)=x(i);
    else
        color(i,3)=abs(x(i));
    end
end

plot(G, 'NodeLabel', G.Nodes.Names,
        'NodeColor', color,
        'MarkerSize', 10 )

%--------Calcolo della dinamica di averaging con nodi stubborn---------
w=W*ones(15,1);
D=diag(w);
P=D\W;

x_curr=x(1:13);         %x(t)
i=1;                    %contatore
MaxIter=100;            %iterazioni massime della simulazione
tol=10e-4;              %Tolleranza come criterio di stop della simulazione
Q=P(1:13,1:13);         %Matrice Q della dinamica
Eu=P(1:13,14:15)*x(14:15); %Fattore Eu che dipende dai nodi stubborn
x_real=(eye(13)-Q)\Eu;  %Calcolo del vettore delle
                        %opinioni limite reale secondo il teorema
x_next=zeros(13,1);     %x(t+1)
err=1;
X=zeros(13,MaxIter);    %Matrice degli stati dei nodi nel tempo
X(:,1)=x_curr;          %Inizializzazione della matrice

while(i<MaxIter && err>tol)
```

```
    x_next=Q*x_curr+Eu; %DINAMICA DI AVERAGING CON NODI STUBBORN

    err=norm(x_next-x_curr)/norm(x_next);

    G.Nodes.x(1:13)=x_next;
    x_curr=x_next;
    i=i+1;

    X(:,i)=x_curr;


    %--------Plot del grafo------------
    color=zeros(15,3);
    for j=1:15
        if G.Nodes.x(j)>0
            color(j,1)=G.Nodes.x(j);
        else
            color(j,3)=abs(G.Nodes.x(j));
        end
    end
    plot(G, 'NodeLabel', G.Nodes.Names,
            'NodeColor', color,
            'MarkerSize', 10 ,
            'NodeFontSize', 10)
    title('Evoluzione delle opinioni nella simulazione della dinamica')
    pause(0.5);
    %------------------------------------
end

%Plot delle traiettorie dei nodi------------------------

X=X(:,1:i); %Riduzione delle dimensioni per salvare spazio
t=1:i
for k=1:13
figure(2)
hold on

plot(t,X(k,:),'Linewidth', 2)

end
title('Traiettorie delle opinioni di ciascun nodo (non stubborn)')
xlabel('Tempo')
ylabel('Opinione')
legend(Names(1:13))

%-----------------------------------------------------------

%Calcolo dell'errore rispetto al valore reale ottenuto dal teorema
```

```
x_sym=G.Nodes.x(1:13)
x_real
err_rel=norm(x_real-x_sym)/norm(x_real)

%Plot del grafo con opinioni reali
G.Nodes.x(1:13)=x_real;

for i=1:15
        if G.Nodes.x(i)>0
            color(i,1)=G.Nodes.x(i);
        else
            color(i,3)=abs(G.Nodes.x(i));
        end
end
    figure(3)
    plot(G, 'NodeLabel', G.Nodes.Names,
            'NodeColor', color,
            'Markersize', 10,
            'NodeFontSize', 10)
    title('Convergenza delle opinioni calcolata direttamente')
```

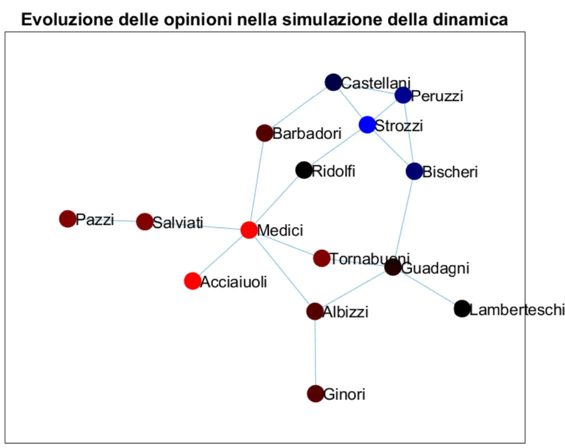



Figure 1: Stato delle opinioni dopo 2 iterazioni

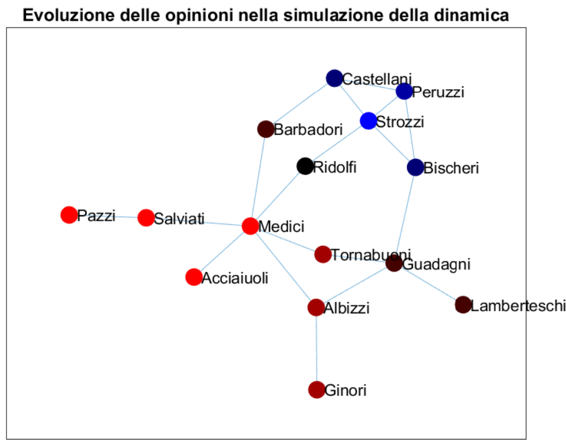



Figure 2: Stato delle opinioni dopo 25 iterazioni

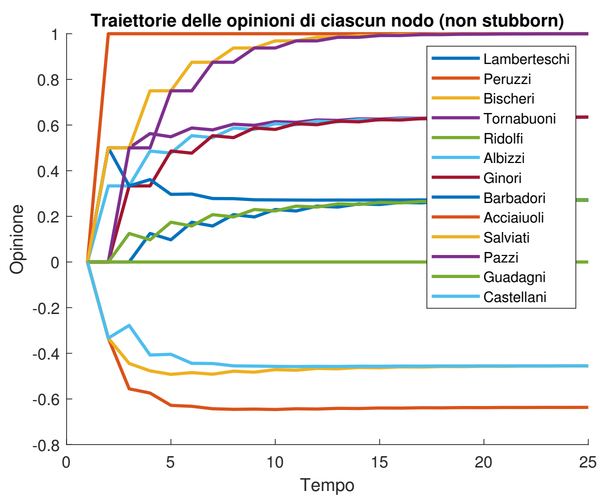
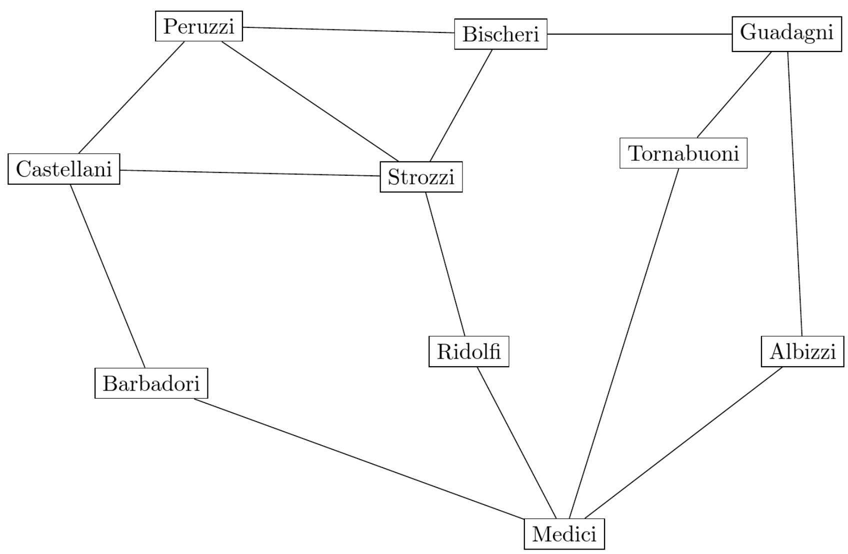



Figure 3: traiettoria delle opinioni

**(c)** Per calcolare analiticamente il vettore di equilibrio della dinamica di averaging con insieme di nodi stubborn dato da

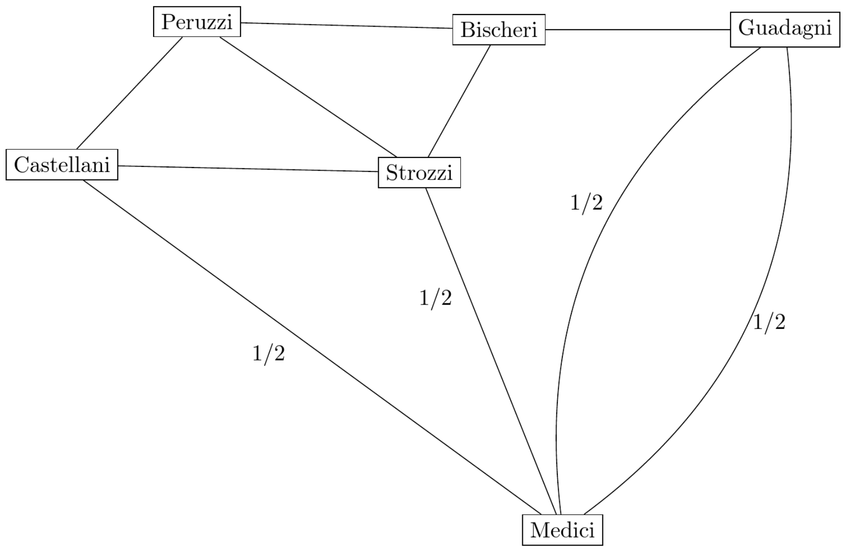
$$\mathcal{S} = \{\text{Medici}, \text{Strozzi}, \text{Castellani}, \text{Guadagni}\}$$

ci si può avvalere dell'analogia con le reti elettriche ed effettuare alcune semplificazioni.
Si può inizialmente andare ad effettuare un operazione di *Glueing* sui nodi che ricevono influenza in maniera univoca da un unico nodo, in particolare:




$$x_{\text{Lamberteschi}} = x_{\text{Guadagni}} \qquad x_{\text{Albizzi}} = x_{\text{Ginori}}$$
$$x_{\text{Salviati}} = x_{\text{Pazzi}} = x_{\text{Medici}} \qquad x_{\text{Acciaiuoli}} = x_{\text{Medici}}$$

Il grafo può quindi essere ulteriormente semplificato andando ad eliminare i nodi $\{\text{Barbadori}, \text{Ridolfi}, \text{Tornabuoni}, \text{Guadagni}\}$ e aggiornando di conseguenza i pesi degli archi:




Risulta quindi evidente che, dal momento che i nodi Peruzzi e Bischeri sono esclusivamente influenzati da nodi stubborn con opinione $u = -1$, necessariamente:

$$x_{\text{Peruzzi}} = x_{\text{Bischeri}} = -1$$

Sfruttando ora l'analogia con la rete elettrica è possibile reintrodurre i nodi eliminati tramite la regola della serie, in particolare per il nodo Barbadori vale:

$$\Phi_{\text{Medici, Castellani}} = \frac{1}{2}\left(x_{\text{Medici}} - x_{\text{Castellani}}\right) = 1 = \Phi$$

Quindi necessariamente

$$\Phi = 1 = 1\left(x_{\text{Medici}} - x_{\text{Barbadori}}\right) \Rightarrow$$
$$\Rightarrow x_{\text{Barbadori}} = 0$$

Analogamente per il resto dei nodi eliminati con la regola della serie vale:

$$x_{\text{Barbadori}} = x_{\text{Ridolfi}} = x_{\text{Tornabuoni}} = x_{\text{Albizzi}} = 0$$

Infine per i nodi eliminati tramite il processo di *glueing* vale:

$$x_{\text{Lamberteschi}} = -1$$
$$x_{\text{Ginori}} = 0$$
$$x_{\text{Acciaiuoli}} = x_{\text{Salviati}} = x_{\text{Pazzi}} = 1$$

**(d)** Il seguente codice si basa sulle dichiarazioni di variabili relative al codice del punto precedente e sull'algoritmo che sfrutta la sub-stocasticità di $(1-\beta)P'$

```
    %Sfruttiamo la proprieta' di (1-beta)P' di essere una matrice
    %sub stocastica per calcolare efficientemente
% il vettore di centralita' Page-Rank

err=1;
beta=0.15
nu=1/15*ones(15,1);
y=ones(15,1);
i=1;
while(i<MaxIter && err>tol)
    y_next=(1-beta)*P'*y+beta*nu;

    err=norm(y_next-y)/norm(y_next);

    y=y_next; i=i+1;
end

err
y
```